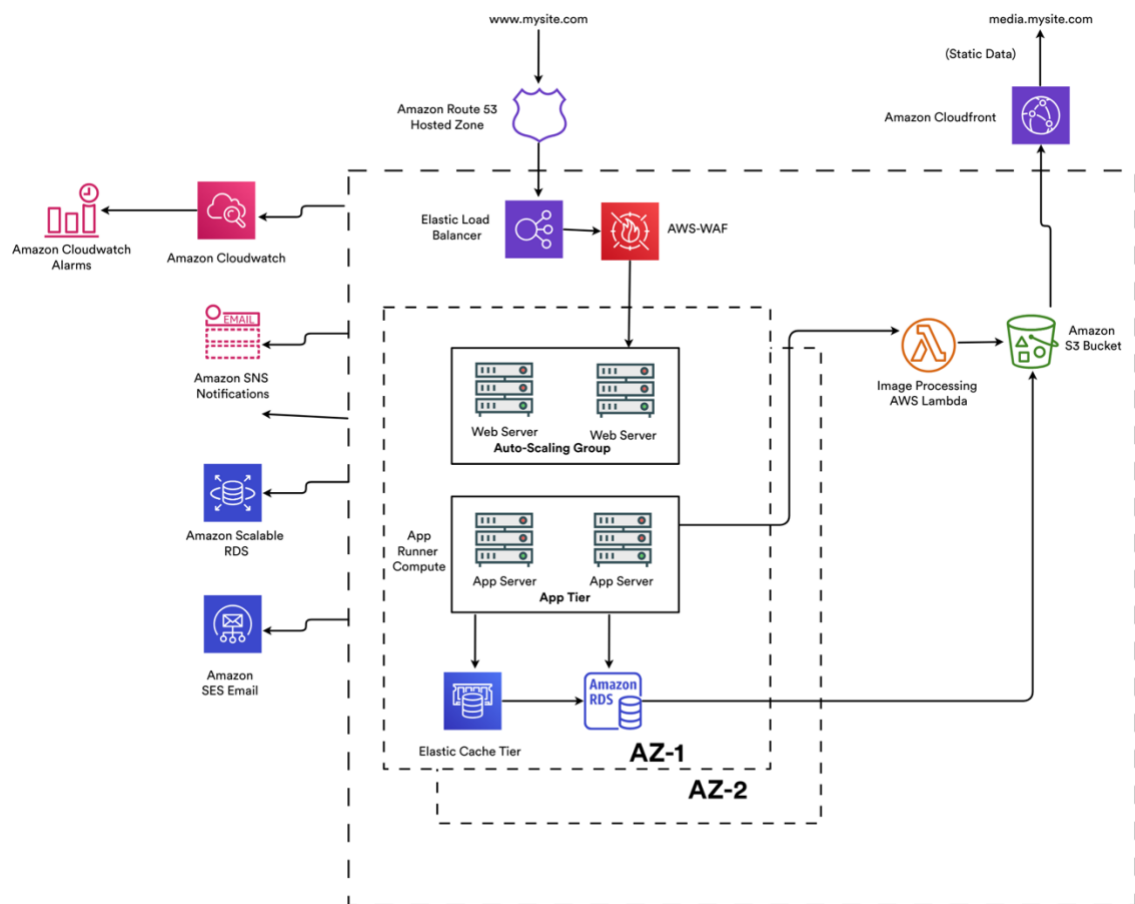


Architecture for the system over the cloud



Design Considerations for Cloud

We will be using the following components:

1. **Amazon Route S3**
It is being used in our application as it is highly available and scalable DNS web service. It is secure and there is complete control over every user within the AWS account.
2. **Elastic Load Balancer**
ELB in our application is responsible for distributing incoming application traffic across multiple targets in one or more availability zones. Our application is present in multiple zones and regions. So, traffic control is handled by elastic load balancer.

3. AWS-WAF

We are using AWS Web Application Firewall in our application for protection against common web exploits and bots that may affect availability, compromise security, or consume excessive resources. We can create multiple security rules to control how the traffic reaches our application.

4. Web Server

We will be serving static content from this server through CDN.

5. App Server

We will be using App runner compute for our app server. Our containerized application will be deployed using app runner and we will have more focus on our application instead of focusing on servers or scaling.

6. Elastic Cache Tier

We will be using Elastic Cache as an in-memory datastore and cache service which will further helps us by improving the performance of web application by retrieving information from managed in-memory caches, instead of relying entirely on slower disk-based databases.

7. Amazon Scalable RDS

We will be using Amazon RDS for storing our data. Amazon RDS supports Auto-Scaling which will scale the storage capacity of our database in response to growing database workloads with zero downtime.

8. AWS Lambda

We will be using AWS lambda for image processing requirement. As mentioned in the requirement, we need a cloud based solutions for image processing of products to ensure the validation of the product being offered on the platform. We will be using AWS lambda for this, which will process the image and upload it on S3.

9. Amazon S3 Bucket

We will be using S3 bucket for storing static data like images/documents/Videos etc.

10. Amazon Cloudfront

Amazon CloudFront is a content delivery network (CDN) service built for high performance, security, and developer convenience. All the static data will be served using this CDN.

11. Amazon Cloudwatch

We will be using Amazon Cloudwatch as a monitoring and observability service. CloudWatch provides us with data and actionable insights to monitor our applications, respond to system-wide performance changes, and optimize resource utilization. CloudWatch collects monitoring and operational data in the form of logs, metrics, and events. We get a unified view of operational health and gain complete visibility of our AWS resources, applications, and services running on AWS and on-premises.

12. Amazon Cloudwatch Alarms

We will be adding cloudwatch alarms to watch metrics and to receive notifications when the metrics fall outside of the levels (high or low thresholds) that we configure.

13. Amazon SNS Notifications

Amazon Simple Notification Service (Amazon SNS) is a fully managed messaging service for both application-to-application (A2A) and application-to-person (A2P) communication.

The A2A pub/sub functionality provides topics for high-throughput, push-based, many-to-many messaging between distributed systems, microservices, and event-driven serverless applications. Using Amazon SNS topics, our publisher systems can fanout messages to many subscriber systems, including Amazon SQS queues, AWS Lambda functions, HTTPS endpoints, and Amazon Kinesis Data Firehose, for parallel processing. The A2P functionality enables us to send messages to users at scale via SMS, mobile push, and email.

14. Amazon SES Email

We will integrate Amazon SES as a cloud email service provider in our application for bulk email sending and we will be paying only for what we use. Amazon SES also supports a variety of deployments including dedicated, shared, or owned IP addresses. It also provides reports on sender statistics and a deliverability dashboard.

Cost estimate (per month and per year) with detailed assumptions

#	Component	Assumptions	Monthly Cost	Yearly Cost
1	Elastic Load Balancing	<p>Region: US East (N. Virginia)</p> <p>Number of Application Load Balancers: 2</p> <p>Average number of requests per second per ALB: 10</p>	32.85 USD	394.2 USD
2	AWS Web Application Firewall (WAF)	<p>Number of Web Access Control Lists (Web ACLs) utilized: 1</p> <p>Number of Rules added per Web ACL: 4</p> <p>Number of Rule Groups per Web ACL: 4</p> <p>Number of Rules inside each Rule Group: 4</p> <p>Number of Managed Rule Groups per Web ACL: 4</p> <p>Number of web requests received across all web ACLs: 1 million per month</p>	33.60 USD	403.2 USD
4	App Runner	High volume production app	25 USD	300 USD
5	Amazon ElastiCache	<p>Nodes: 2</p> <p>Instance type: cache.t2.small</p> <p>Utilization: 50% per month</p> <p>Cache Engine: Memcached</p> <p>Cache Node Type: Standard</p>	24.82 USD	297.84 USD
6	Amazon RDS for Oracle	<p>Number of RDS for Oracle instances: 1</p> <p>Instance type: db.m3.large</p>	158.05 USD	1896.6 USD

		Utilization: 50% per month Deployment option: Multi-AZ Database edition: Enterprise Storage for each RDS instance: General Purpose SSD (gp2) Storage amount: 100gb		
7	AWS Lambda	Service settings Region: US East (N. Virginia) Architecture: x86 Number of requests: 1000000/month Duration of each request (in ms): 10ms Amount of memory allocated: 10gb Lambda@Edge settings Number of requests: 1000000/month Duration of each request (in ms): 10ms Amount of memory allocated: 10gb	5.60 USD	67.2 USD
8	Amazon Simple Storage Service (S3)	S3 Standard storage: 10gb/month PUT, COPY, POST, LIST requests to S3 Standard: 100000/ month GET, SELECT, and all other requests from S3 Standard: 100000/month Data returned by S3 Select : 100gb/month	0.84 USD	10.08 USD
9	Amazon Cloudfront	Data transfer out to internet : within 1024 gb Data transfer out to origin : 10gb/month Number of requests (HTTPS) : within 10 million free requests	0.20 USD	2.4 USD
10	Amazon CloudWatch	Metrics Number of Metrics (includes detailed and custom metrics)	4.58 USD	54.96 USD

		: 10 APIs GetMetricData: Number of metrics requested: 1000 GetMetricWidgetImage: Number of metrics requested: 1000 Number of other API requests: 10000 Standard Logs: Data Ingested : 1gb Expected Logs Data scanned GB: 10gb Number of Dashboards: 3 Number of Standard Resolution Alarm Metrics: 3 Number of High Resolution Alarm Metrics: 2		
11	Amazon Simple Notification Service (SNS)	Region: US East (N. Virginia) Requests: 1 million per month HTTP/HTTPS Notifications: 100000 per month EMAIL/EMAIL-JSON Notifications: 100000 per month Mobile Push Notifications: 100000 per month	2.03 USD	24.36 USD
12	Amazon Simple Email Service (SES)	Email messages sent from email client: 10000 Email messages received: 1000 Average size of email received: 100kb	1.10 USD	13.2 USD
TOTAL			288.67 USD	3464.04 USD